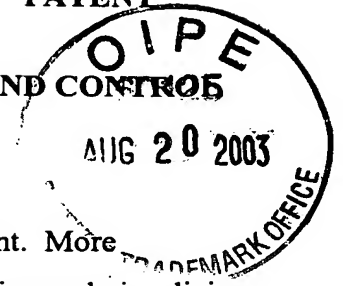


REAL TIME MONITORING MANUFACTURING SCHEDULING AND CONTROL**FIELD OF THE INVENTION**

This invention relates in general to the field of information management. More particularly, the invention relates to a system and method of real-time monitoring and visualizing of manufacturing processes and other key business processes.

BACKGROUND OF THE INVENTION

Managing daily business activities efficiently is very important to keep operating costs low and customers satisfied. In today's business world, it is very difficult and complicated to obtain real-time information necessary to manage business processes and monitor assets. Using incomplete or inaccurate data can lead to strategic and tactical mistakes, long lead times, high work in progress (WIP), and quality problems. These problems cost businesses time and money due to higher capital expenses, decreased cash flow through lower inventory turnover, higher safety/buffer stocks, and decreased availability.

Conventional systems do not provide an efficient, flexible and reliable system to monitor the daily, and short and long term activities of an enterprise in real-time. Thus, in view of the foregoing, there is a need for systems and methods that overcome the limitations and drawbacks of the prior art. In particular, there is a need for system that can monitor the activities of an enterprise in real-time to address the limitations of the prior art and provide decision makers with the information they need. The present invention provides such a solution.

SUMMARY OF THE INVENTION

The present invention provides a system for visually displaying real-time enterprise management information. In accordance with an aspect of the invention, there is provided systems and methods for visually displaying real-time enterprise status information related to a status of jobs running on machines. The systems and methods include an application integration platform that receives the real-time enterprise status information and analyzes the information to determine at least one key performance indicator, a process control server that receives status information from at least one work center and forwards the status information to the application integration platform, a database containing information related to manufacturing processes

performed at the at least one work center, and a graphical user interface that interfaces with the application integration platform to provide a visual display of the at least one the key performance indicator. The system provides for minimizing the number of tardy jobs running on machines within the at least one work center.

In accordance with a feature of the invention, jobs are initially scheduled on the machines and the system provides for control of scheduling after jobs are released to the work center shop floor. Jobs may be rescheduled from one machine to another in accordance with an expected completion time. The expected completion time is determined based on an average completion time for a set of jobs assigned to the machines plus a total variation for all jobs in the machines.

The system may graphically present scheduled jobs for each machine and an expected completion time, wherein the expected completion time includes any tardiness. A maximum number of tardy jobs is defined as:

$$\begin{aligned} & \text{Max}\{d_i - E_m < 0\} \\ & i \in K \\ & m \in M \end{aligned}$$

wherein

$$E_m = T_c + \sum_{i=1}^k \mu_i + z_{\alpha/2} \sigma_{1..k}$$

and wherein

d_i = Due date for job i

E_m = Expected completion time of the jobs on machine m

M = Machine set

K = Set of jobs scheduled on selected machine

z_{α} = Standard normal distribution coefficient for given risk, α

σ_i = Standard deviation for a particular job in machine M

μ_i = Average duration for a particular job in machine M

T_c = Current time.

As expected completion time for a set of jobs may defined as:

$$\text{Min}_{j=1}^M \left\{ T_c + \sum_{i=1}^k \mu_i + z_{\alpha/2} \sigma_{mT} \right\}$$

where

$$\sigma_{mT} = SQRT(\sigma_1 + \sigma_2 + \dots + \sigma_n)$$

Where

M = Machine set

K = Set of jobs scheduled on selected machine

5 z_α = Standard normal distribution coefficient for given risk, α

σ_{mT} = Total standard deviation for set of jobs in machine m

μ_i = Average duration for a particular job in machine M

T_c = Current time

10 In accordance with another aspect of the invention, an uncertainty level is assigned based on an expected completion time for jobs, the uncertainty time being a time within which it is expected that a predetermined percentage of the jobs will be completed.

Additional features and advantages of the invention will be made apparent from the following detailed description of illustrative embodiments that proceeds with reference to the
15 accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing summary, as well as the following detailed description of preferred embodiments, is better understood when read in conjunction with the appended drawings. For
20 the purpose of illustrating the invention, there is shown in the drawings exemplary constructions of the invention; however, the invention is not limited to the specific methods and instrumentalities disclosed. In the drawings:

Fig. 1 is a block diagram showing an exemplary computing environment in which aspects of the invention may be implemented;

25 Figs. 2-6 illustrate the architecture and components of the present invention;

Fig. 7 illustrates the various levels of detail and information that may be provided to different classes end-users within an enterprise;

Figs. 8-15 illustrate exemplary user interfaces and output reports provided by the present invention; and

30 Figs. 16-17 illustrate scheduled jobs on various machines in a manufacturing facility.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

Given the complex manufacturing systems and market conditions, scheduling has a critical impact on a company's performance. Moreover, after a schedule is made and released to the shop floor, assessing the manufacturing performance in a shop floor in a timely matter is very difficult. Even if jobs are scheduled optimally, after they are released the shop floor, there may be delays in throughput. Throughout the production cycle there will be several reasons which will prevent the jobs from being completed on time. For example, a released job might have different priority due to a customer request. In this case, one should be able to evaluate best options to finish the job on time to satisfy the customer. Reasons such as missing material, machine breakdown, wrong Bills of Material, absenteeism, etc., make it very difficult to control and monitor in the shop floor to prevent late deliveries.

The present invention is directed to systems and methods for monitoring whole enterprise processes and key performance indicators (KPI) both locally and remotely. Classes of users (e.g., managers, engineers, operators, etc.) are able to monitor KPIs, such as financial indicators, market activities, overall company conditions, throughput, available capacity, machine status, quality information, etc. Decisions can be made on day-to-day activities, and short and long term activities based on the monitored results. The present invention includes software that is based on an object-oriented architecture to provide a platform that can be easily configured and scaled to increase functionality and enterprise growth.

The methodology described herein is a robust and easy way to assess a scheduling status of jobs in the shop floor in real time and change the job scheduling if necessary.

Exemplary Computing Environment

Fig. 1 illustrates an example of a suitable computing system environment 100 in which the invention may be implemented. The computing system environment 100 is only one example of a suitable computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing environment 100 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment 100.

The invention is operational with numerous other general purpose or special purpose computing system environments or configurations. Examples of well known computing systems, environments, and/or configurations that may be suitable for use with the invention

include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

5 The invention may be described in the general context of computer-executable instructions, such as program modules, being executed by a computer. Generally, program modules include routines, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices
10 that are linked through a communications network or other data transmission medium. In a distributed computing environment, program modules and other data may be located in both local and remote computer storage media including memory storage devices.

 With reference to Fig. 1, an exemplary system for implementing the invention includes a general purpose computing device in the form of a computer 110. Components of computer 110
15 may include, but are not limited to, a processing unit 120, a system memory 130, and a system bus 121 that couples various system components including the system memory to the processing unit 120. The system bus 121 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. By way of example, and not limitation, such architectures include Industry
20 Standard Architecture (ISA) bus, Micro Channel Architecture (MCA) bus, Enhanced ISA (EISA) bus, Video Electronics Standards Association (VESA) local bus, and Peripheral Component Interconnect (PCI) bus (also known as Mezzanine bus).

 Computer 110 typically includes a variety of computer readable media. Computer readable media can be any available media that can be accessed by computer 110 and includes
25 both volatile and non-volatile media, removable and non-removable media. By way of example, and not limitation, computer readable media may comprise computer storage media and communication media. Computer storage media includes both volatile and non-volatile, removable and non-removable media implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules or other
30 data. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other optical

disk storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by computer 110. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term "modulated data signal" means a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared and other wireless media.

Combinations of any of the above should also be included within the scope of computer readable media.

The system memory 130 includes computer storage media in the form of volatile and/or non-volatile memory such as ROM 131 and RAM 132. A basic input/output system 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110, such as during start-up, is typically stored in ROM 131. RAM 132 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120. By way of example, and not limitation, Fig. 1 illustrates operating system 134, application programs 135, other program modules 136, and program data 137.

The computer 110 may also include other removable/non-removable, volatile/non-volatile computer storage media. By way of example only, Fig. 1 illustrates a hard disk drive 140 that reads from or writes to non-removable, non-volatile magnetic media, a magnetic disk drive 151 that reads from or writes to a removable, non-volatile magnetic disk 152, and an optical disk drive 155 that reads from or writes to a removable, non-volatile optical disk 156, such as a CD-ROM or other optical media. Other removable/non-removable, volatile/non-volatile computer storage media that can be used in the exemplary operating environment include, but are not limited to, magnetic tape cassettes, flash memory cards, digital versatile disks, digital video tape, solid state RAM, solid state ROM, and the like. The hard disk drive 141 is typically connected to the system bus 121 through a non-removable memory interface such as interface 140, and magnetic disk drive 151 and optical disk drive 155 are typically connected to the system bus 121 by a removable memory interface, such as interface 150.

The drives and their associated computer storage media, discussed above and illustrated in Fig. 1, provide storage of computer readable instructions, data structures, program modules and other data for the computer 110. In Fig. 1, for example, hard disk drive 141 is illustrated as storing operating system 144, application programs 145, other program modules 146, and program data 147. Note that these components can either be the same as or different from operating system 134, application programs 135, other program modules 136, and program data 137. Operating system 144, application programs 145, other program modules 146, and program data 147 are given different numbers here to illustrate that, at a minimum, they are different copies. A user may enter commands and information into the computer 20 through input devices such as a keyboard 162 and pointing device 161, commonly referred to as a mouse, trackball or touch pad. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 120 through a user input interface 160 that is coupled to the system bus, but may be connected by other interface and bus structures, such as a parallel port, game port or a universal serial bus (USB). A monitor 191 or other type of display device is also connected to the system bus 121 via an interface, such as a video interface 190. In addition to the monitor, computers may also include other peripheral output devices such as speakers 197 and printer 196, which may be connected through an output peripheral interface 190.

The computer 110 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 180. The remote computer 180 may be a personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the computer 110, although only a memory storage device 181 has been illustrated in Fig. 1. The logical connections depicted include a local area network (LAN) 171 and a wide area network (WAN) 173, but may also include other networks. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

When used in a LAN networking environment, the computer 110 is connected to the LAN 171 through a network interface or adapter 170. When used in a WAN networking environment, the computer 110 typically includes a modem 172 or other means for establishing communications over the WAN 173, such as the Internet. The modem 172, which may be internal or external, may be connected to the system bus 121 via the user input interface 160, or

other appropriate mechanism. In a networked environment, program modules depicted relative to the computer 110, or portions thereof, may be stored in the remote memory storage device. By way of example, and not limitation, Fig. 1 illustrates remote application programs 185 as residing on memory device 181. It will be appreciated that the network connections shown are
5 exemplary and other means of establishing a communications link between the computers may be used.

Exemplary Distributed Computing Framework and Architecture

The present invention is directed to systems and methods for monitoring whole enterprise processes and key performance indicators (KPI) that provides for decoupling of functionality
10 into individual, standalone, reusable subsystems. The present invention provides small, simple interfaces between components, and similarity of concepts within the architecture. The decomposition of the system into reusable components is performed in such a way, that the system can be adapted to changes in requirements via an exchange of a minimal set of components. To accomplish these goals, the present invention is preferably implemented using
15 VB coding standards.

Referring to Figs. 2-6, the present invention integrates the disparate manufacturing and information systems into an Application Integrator Platform (AIP) platform 200. The AIP platform 200 may be implemented on a computer similar to computer 110 and receive data from several systems and to provide visual monitoring. The AIP is platform independent, and may be
20 implemented using VB coding standards, JAVA, or within a .Net Environment. If the platform 200 needs financial information, the data may be extracted from an Enterprise Resource Planning (ERP) system 220 via an exchange of XML data 230 in real-time (see, Fig. 3). The common visualization program enables the sharing of business content and data across the enterprise by supporting integration to Enterprise Resource Planning (ERP) system 220 via an exchange of
25 XML data 230 in real-time (see, Fig. 3) for financial information or to other enterprise applications such as Quality, Manufacturing, Execution System (MES) 208, Product Data Management (PDM) 212, etc. The ERP system 220 preferably comprises R/3 release 4.6C, available from SAP AG. The MES 208 and PDM 212 may comprise Lotus Notes, available from IBM Corporation.

30 Users interact with the system by logging on to the AIP 200 locally, or remotely using a WWW interface via the Internet. Preferably, two interfaces are implemented, an AIP Thin Client

204 for the BA view, and an AIP Client 206 for the BAU view. The BAU shows a Business Area Unit (e.g., a manufacturing facility or business), whereas the BA view shows a Business Area (a group of common BAUs). The AIP Thin Client 204 and AIP Client 206 may be implemented on, e.g., computer 110 to connect to various components via a LAN 236 and/or corporate intranet
5 240. In addition, other components may communicate with each other via the LAN 236 or wireless access points 238 and bridges 239. Business activity can be graphically monitored by the system in real-time to allow users to react and make critical business decisions based on performance information.

Manufacturing status data 214 may be gathered via a barcoding system or machine
10 sensors and/or other controls from, e.g., a coil winding machine 216, core stacking machine 218, drying/filling machine 220, active part assembly machine 222, tank/final assembly machine 224 and/or test machine 226. It is noted that the present invention is not limited to such machines and/or data, and may be utilized to capture and analyze data from other types of machines and information sources. Barcode data gathering techniques are well known to those of ordinary skill
15 in the art and equipment therefor is available from, e.g., Symbol Technologies, Holtsville, NY.

Figs. 3-5 illustrate architecture of Fig. 2 with particular reference to the Coil Winding machine 216. The winding machine 216 may be monitored via well known sensors to track machine status (e.g., on/off) and state (e.g., turning). Data related with manufacturing may be gathered from feedback points in the plants using sensors and the barcode system (including
20 database 228). Data signals will be sent directly to an OLE from Process Control (OPC) server 202.

Those of ordinary skill in the art will understand that OPC is a series of standards specifications. The first standard (called the OPC Specification and now called the Data Access Specification) resulted from the collaboration of a number of leading worldwide automation
25 suppliers working in cooperation with Microsoft Corp. Originally based on Microsoft's OLE COM (component object model) and DCOM (distributed component object model) technologies, the specification defined a standard set of objects, interfaces and methods for use in process control and manufacturing automation applications to facilitate interoperability. The COM/DCOM technologies provided the framework for software products to be developed.
30 There are now hundreds of OPC Data Access servers and clients.

When sensor signals are sent to the OPC Server 202, the Aspect Integrator Platform (AIP) 200 or Web Interface will be used to gather the data and display the information to the end users. This data will include operator activities from the data feedback points in the manufacturing plant. Manual machines 232 may have optical and voltage sensors to measure status of the manufacturing, which are sent to a controller 233 for transmission to the OPC Server (OLE for Process Control) 202. For highly automated machines 234, which have the built in controller, the signals are sent directly to the OPC server 202. The AIP 200 may extend across a single or multiple disparate locations to extent the value chain and enable real-time correlation.

As shown in Fig. 6, there is the test oven 220 as monitored by the present invention. The oven may be monitored via well known sensors to track temperature, vacuum, etc. and the data therefrom communicated via the wireless access point 238 and the wireless bridges 239 to the AIP platform 200. Barcoding techniques may also be used to track activities via SAP data and ERP System 220. As is now evident to those of ordinary skill in the art, any performance characteristic of any machine on the shop floor may be monitored by the present invention by collecting data therefrom.

In accordance with the present invention, the following exemplary key performance indicators (KPI) may be monitored by the API 200 of the present invention:

1. Throughput time (days)

Throughput time may be calculated for overall manufacturing and for each process as days spent in production for a project name, unit, and work order:

Throughput time = Completion time - start time (for work order, project name, and unit).

In accordance with the present invention, users may calculate a throughput time for given dates, project name, work order, and unit. For example, if a user wants to see all project throughput times for last 30 days, the user will be able to view a project name and its associated throughput times. If he/she wants to see a particular project or unit throughput time, he/she should be able to see the project name or unit throughput time as a whole or for individual work centers.

2. Manufacturing hours (total and by activity)

Manufacturing time may be calculated for each activity in the shop floor as hours spent on a project, work order, and unit. This KPI is similar to throughput time, except that real

manufacturing hours is preferably tracked on the shop floor. Manufacturing activities may be defined as, e.g., set up, processing, and set out. Similarly, manufacturing problems may be defined as, e.g., break down, missing parts, quality issues, operator breaks, and technical clarification.

5 In accordance with the present invention:

Manufacturing time (overall) = end time (or current time) - start time - all problem times (e.g., employee break, technical problems, etc.)

10 Manufacturing time (for an activity) = end time (or current time) for an activity - start time for the activity - all problem times (e.g., employee break, technical problems, etc.). The present invention gathers information for given work center (e.g., activities and problem time), project name, and unit. Similarly, same time calculation should be performed for manufacturing problems. It is preferable that the user is able to obtain problem codes and times for break down, missing parts, quality issues, operator breaks, and technical clarification. For example, the user should be able to obtain break down times for all projects for a particular work center or all work
15 centers for a given date interval.

3. Work Center Utilization

In accordance with the present invention, work center utilization is a measure of the actual productive time for a work center:

Utilization for a machine = (productive time on the machine) / (endtime - start time).

20 Machine productive hours are preferably extracted through sensor data. Other required times may come from a database storing such information. It is preferable to obtain labor hours associated with particular work center through an employee ID to calculate total hours spent on the machine. The present invention may obtain these data for a given date and work center. If a user wants to view average work center utilization (i.e., not a particular machine utilization), the
25 present invention preferably calculates average utilization after obtaining machine utilization.

4. Man-hour capacity report

Man-hour utilization is similar to the machine utilization. The present invention calculates total labor hours for each work center as follows:

Man hour utilization = (used man hour) / (total available man hour)

Used man-hour will is preferably calculated in a similar manner as in the work center utilization (KPI 3) above. Total man-hours may be a manual entry and may be determined for user-specified dates.

5. Planned vs. Actual

5 As part of the project tracking purposes, it is preferable to calculate the following measures with scheduling: the difference between planned start time and actual start time, the difference between planned completion and actual completion time, and the difference between planned hours and actual hours.

10 Planned start and completion times for a work order are defined as the original scheduling times. Actual start time and completion time may be extracted from, e.g., a database 228. The present invention provides the user with a view of planned vs. actual time for each work station for a given project name. Users may also want to track planned vs. actual for each project for a given work center. For example, if he or she wants to see all the projects status for a winding center, he or she should be able to list all projects with planned vs. actual time in the
15 work center.

6. Work in Process (WIP)

WIP is the number of units in each work center either waiting or processing for a given time. It is often necessary to find out how many units are waiting or processing in the work stations. In exemplary manufacturing environment 214 of the present invention, these are: a
20 number of units in the winding process, a number of cores waiting (i.e., arrived, but not processed, thus are considered raw material), a number of active parts waiting (i.e., finished, but into the next process), a number of complete units waiting for final assembly (i.e., finished, but into the next process), a number of complete units waiting for final testing, a material weight for drums in the winding, and a number of tanks in the system.

25 The present invention provides WIP quantity by project name, unit, and work order for each station. For example, if user wants to see WIP quantity in the winding machine, all projects, units, and work orders assigned to winding work center are returned.

Additional KPIs may be defined in accordance with the particular needs of the enterprise manufacturing environment.

30 The database 228 of the present invention will now be described. Data storage and replication in database 228 is preferably implemented using SQL Server 2000, available from

Microsoft Corp., Redmond, WA. In the present invention, directory replication and database replication is used for performance optimization. Data is preferably stored and managed locally. The BAU level the relevant KPI are summarized and replicated daily (see, Fig. 5). From the BAU data, the records and aspects are replicated to the BA server (see, Fig. 4) for viewing. The database replication is a functionality of SQL.

Processing at each workstation is maintained by scanning barcodes (using barcode scanner 242) representative of a particular status at the workstation. The data is stored in the database 228 for processing of the KPIs discussed above. Table One, below, relates the KPI monitoring data requirements to determine the KPI value.

Monitoring Requirements	Data needed
KPI: Throughput time (days) Overall manufacturing throughput Throughput for each processes Throughput = (end time-start time)	Work Order (WO) number & related operations numbers Work Station ID Start time for WO (at each station) End time for WO (at each station)
KPI: Throughput activity analysis: Set up Processing Set out Break down Missing parts Quality issues Operator breaks, etc.	WO number (at each station) Activity code and/or problem code Start time for WO (at each station) End time for WO (at each station)
KPI: Utilization: Machine used "on/off/break down" Machine turning "yes/no" Capacity Utilization = (Man-hour)/ (theoretical labor capacity)	Work station ID Activity code WO number Employee Information (e.g., badge number)
KPI: Planned vs. Actual: Difference between planned start time and actual start time. Difference between planned completion and actual completion time. Difference between planned hours and actual hours.	WO number (at each work station) Planned start time Actual start time Planned completion time Actual completion Time

Monitoring Requirements	Data needed
KPI: WIP for the each activity: Units in the winding process. Number of cores waiting. Number of active part waiting. Number of complete unit waiting for final assembly. Number of complete unit waiting for final testing. Number of drums. Number of tanks in the system.	WO number (at each work center) Work station ID Start time for WO End time for WO Material code
KPI: Non conformity reports Number of "launched" weekly Different status ("launched", "on treatment", all excepted "solved") Cumulated number of manufacturing hours lost	Work Center SAP mapping (to get the product code) Material code WO number
Test reports for the final testing for each unit.	Test Reports Work station ID
Order Tracking: Which unit/project at which station.	WO number (at each station) Work Station ID

Table One

As noted above, in accordance with the present invention, data is captured using a barcode system 242 to determine and display the KPIs noted above. The structure of data items, such as databases, OPC servers, and configuration data will now be described. Table Two below defines an exemplary barcode system and type of the data collected or updated during the manufacturing operations. The scope of the claims of the present invention shall not be limited by the exemplary system described below, as other events and processes may be captured by the system of the present invention.

A data collection event, as used herein, is the manufacturing activity that will occur in a particular machine or work center. During these events, data related to the shop floor is collected and updated. For example, when operator starts to work, he or she scans or enters his or her badge number, WO number, and activity code. This activates the work process in the particular workstation and the data will be sent to the database 228 until the next event, which may be, e.g., an "end work," "pause," etc. event. In accordance with the present invention, the AIP 200 may access the database 228 to obtain the information in Table Two for presentation to a user.

	Start Work	End Work	Pause for Regular Breaks	Stop Work for a Problem	Quantity Report
Procedure	1. Scan badge and WO 2. Enter activity code and press enter	1. Scan badge and press enter	1. Scan badge and WO 2. Enter problem code	1. Scan badge and WO 2. Enter problem code 3. Press enter	1. Scan badge and WO 2. Enter the quantity of completed 3. Enter scrap quantity
Data Needs to be captured through barcode system	1. Badge Number 2. WO number 3. Product ID 4. Operation ID 5. Start time for WO 6. Status of WO 7. Material code 8. Material quantity 9. Activity Code 10. Work station ID 11. Work Center ID	1. Badge Number 2. WO number 3. Product ID 4. Operation ID 5. End time for WO	1. Badge Number 2. WO number 3. Product ID 4. Operation ID 5. End time for WO	1. Badge Number 2. WO number 3. Product ID 4. Operation ID 5. End time for WO 6. Problem code	1. Badge Number 2. WO number 3. Product ID 4. Operation ID 5. End time for WO 6. Quantity completed 7. Scrap quantity 8. Status of WO
Other data related to the event	1. Project ID 2. Planned start time 3. Actual start time 4. Customer order number 5. Activity descriptions			1. Problem descriptions	1. Project ID 2. Planned completion time 3. Actual completion time 4. Customer order number

Table Two

In Table Two above, the following definitions apply:

WO number: This is a number that uniquely identifies a work order that was released to shop floor to the particular workstations/work centers.

5 Product ID: Identifies a unique product number that was assigned by BAU.

Operation ID: Defines the detailed work instructions for a particular work order. These may be assembly instructions and/or component list.

10 Work Center ID: Work centers are group of machines that was identified by plant. They are usually logically grouped machines such as winding machines or assembly stations. The work center ID is the number that uniquely identifies the group. Typically, work centers are same as in ERP systems work centers.

Work Station ID: The ID of a particular machine in a work center.

Start time for WO: This indicates that work for a particular job, by a particular operator has been initiated. When the operator starts or resumes the work order, this data is updated in the database. As operator begins working in the workstation, he/she scans his/her employee ID barcode and then a barcode on the WO paper, and then finally enters an "activity code."

5 End time for WO: When the operator finishes the work for that job, he/she scans the same barcodes noted above, and enters the activity code to end the job. When the operator ends the work for a work order, the end time is captured. It is preferable to capture and update this data in the database each time the operator stops or ends the work order.

10 Status of the work order: This data is used to trace the status of the work order. It preferably contains three statuses: "completed," "not started," and "started." When the WO is in the system (e.g., the database 228), it is assigned the status of "not started." When the operator starts to work on the work order, the status is updated to "started." Finally, when the work is completed, the status is updated to "completed."

15 Activity code: These define predetermined activities such as set up, processing, set out, etc. These codes are used to extract productive time in the shop floor.

Activity descriptions: These are the descriptions of the activity codes.

Problem codes: These are the codes that define predetermined problems such as material missing, machine breakdown, etc. These will be used to extract non-productive time in the shop floor to help point out improvement areas to the users.

20 Problem descriptions: These are the codes that define predetermined problems, such as material missing, machine breakdown, etc. These will be used to extract non-productive time in the shop floor. It will help the user to determine improvement areas.

Badge number: uniquely identifies an employee number.

Planned start time: This is the start time that was scheduled by the plant.

25 Actual start time: This is the actual start time for a particular work order. It is often different than the planned start time.

Planned completion time: This is the end time that was scheduled by the plant.

Actual completion time: This is the actual end time for a particular work order. It is often different than the planned end time.

Material code: the material code uniquely identifies a particular material or sub assembly in the work center. It may be, e.g., a number of coil drums or tanks that will be used in the stations.

5 Material quantity: It will be amount of material in particular workstations. Since these data will help to determine the WIP between the stations, it is preferably captured when material enter or leave the work stations

Customer Order Number: A unique customer order number assigned by plant. This number is preferably tied to a work order and product ID.

Project number: These define that a unique project belongs to a particular customer.

10 Quantity completed: When operator finishes a work order, he or she reports the quantity completed. This field shows the completed quantity of the WO.

Scrap quantity: These are the quantity reported to the system by operators.

Referring now to Fig. 7, there is illustrated the various levels of information that the present invention may present to different levels of users. For example, top management may be
15 provided financial and other critical KPI information, whereas middle management may be provided analysis capabilities for on-going activities. Further downstream, supervisors and operators may be provided with KPI information for their immediate areas and routing/scheduling information for a selected period of time. Thus, the present invention can provide all levels of employees with information specifically tailored for their needs. This
20 advantageously reduces extraneous information and increases the relevance of the data provided to the user such that critical decisions can be made in a timely fashion. Table Three, below, further illustrates the various levels of detail that may be provided.

User	Key Performance Measures (KPIs)	Subject Matter Experience	Technology Experience	Other Attributes
BA Manager	<ul style="list-style-type: none"> Throughput time Throughput activity analysis Machine Utilization (on/off, turning/not turning, and capacity utilization for windings, oil, and filling) 	Knowledgeable of business	Novice	Read English
BAU Manager	<ul style="list-style-type: none"> Throughput time Machine Utilization (on/off, turning/not turning, and capacity utilization for windings, oil and filling) Throughput activity analysis WIP levels throughout the plant for key components Non conformity reports Manufacturing work order status 	Knowledgeable of business and manufacturing process	Novice	Read English
Engineers/ Supervisors	<ul style="list-style-type: none"> Throughput time Machine Utilization (on/off, turning/not turning, and capacity utilization for windings, oil, and filling) Throughput activity analysis WIP levels throughout the plant for key components Non conformity reports <ul style="list-style-type: none"> Manufacturing work order status 	Knowledgeable of business and manufacturing process	Novice	Read English
Operators	<ul style="list-style-type: none"> Throughput time Throughput activity analysis 	Knowledgeable of manufacturing process	Novice	Read English

Referring now to Figs. 8-15, there are illustrated several exemplary graphical user interfaces that are provided to the user to query various KPIs and output results with respect to various machines, WOs, and facilities. In the interfaces, the user may select from the following

5 criteria to obtain KPI information: employees, periods of time, dates, work centers, work

stations, etc. to obtain enterprise performance information. The user interfaces are not limited to those illustrated in Figs. 8-15 as other information related to the shop floor may be provided to the user.

Referring now to Fig. 16, there is illustrated a number of jobs running on several machines. In accordance with the present invention, a method for minimizing the number of tardy jobs is disclosed herein. A preferred method for initial scheduling is the Earliest Due Date (EDD) method, however, any method for initial scheduling may be implemented. The present invention provides a method to control the scheduling status after a job is released to the shop floor. Thus, after a schedule is created and released to the shop floor, the present invention provides the user with a way to follow up the execution of the schedule visually.

As illustrated in Figs 8-15, the user may be provided with a visual indication of the difference between actual and planned schedules. For example, the user may check a particular job in the system, and reschedule the job by assigning it to a different machine if available, or continue with the current scheduling. To check for tardy jobs, the user opens an AIP scheduling screen, which consists of parameter selections and graphs to view tardy jobs. The user can select the tardiest job or a job that may have a new priority due to changed conditions (for example, customer priority).

Referring again to Fig. 16, the system shows scheduled jobs for each machine with the expected completion time. The tardiness for each job shown as a hatched portion. The user can visually determine the tardiest job or let the system determine which job is the tardiest. In the exemplary graphs, there is no tardiness for job k in machine 1. On the other hand, job k in machine 2 has tardiness.

In accordance with the present invention, the maximum tardy jobs defined as:

$$\begin{aligned} & \text{Max}\{d_i - E_m < 0\} \\ & i \in K \\ & m \in M \end{aligned}$$

Where

$$E_m = T_c + \sum_{i=1}^k \mu_i + Z_{\alpha/2} \sigma_{1..k}$$

Where

- 30 d_i = Due date for job i
 E_m = Expected completion time of the jobs on machine m

M = Machine set

K = Set of jobs scheduled on selected machine

z_α = Standard normal distribution coefficient for given risk, α

σ_i = Standard deviation for a particular job in machine M

5 μ_i = Average duration for a particular job in machine M

T_c = Current time

The preferred measure for decision making is the expected completion time for a particular job. The measure uses the average completion time plus the expected deviation for whole set of jobs. The following formula provides the expected completion time for each selected job k in each machine:

$$\text{Min}_{j=1}^M \left\{ T_c + \sum_{i=1}^k \mu_i + z_{\alpha/2} \sigma_{mT} \right\}$$

where

15 $\sigma_{mT} = SQRT(\sigma_1 + \sigma_2 + \dots + \sigma_n)$

Where

M = Machine set

K = Set of jobs scheduled on selected machine

z_α = Standard normal distribution coefficient for given risk, α

20 σ_{mT} = Total standard deviation for set of jobs in machine m

μ_i = Average duration for a particular job in machine M

T_c = Current time

If the user wants to reschedule a job to another machine, he or she needs to check how the tardiness will change by reassigning the job to a different machine. Fig. 17 shows job k on the different machines with the tardiness. First, expected completion time for previous jobs that was assigned to a particular machine is calculated, and then expected time of the selected job added using the formulas specified above. The expected completion time will be calculated using average completion time for set of jobs assigned to the machine jobs plus total variation for the all jobs in the machines. Using the graph one can look the expected completion time and

reassign the job to another machine. The user may check the expected completion times and pick a particular machine that will finish the job in earliest time possible. For example, since selected job k has no tardiness in the second machine, the user optimally reassigns the job to the second machine. Other machines have tardiness as we see in the Fig. 17.

5 Since changing a job assignment to a different machine will change all the calculations described above, the user continues the reassignment process until he or she is satisfied with the results. This process will can be continued by selecting next tardy job and assign it to a different machine. The user also may stop this process until all jobs checked. This process can be done at regular intervals or different intervals based on the needs of the user. This will be depend on the
10 shop floor status and how the changes and uncertainty will affect the job scheduling. As an example, after a scheduling job is completed, a missing material may postpone a job in a particular machine. Then user then assesses the situation using the methodology above. It is also possible to simulate the different options with a calculated risk before the jobs released to the shop floor. To perform this, the user changes the standard deviation to a desired level for risk.

15 As is now evident to those of ordinary skill in the art, the methodology above takes into consideration uncertainty due to activities of shop floor, customer priorities, market conditions, etc. Uncertainty is considered using a normal distribution as a base model. The user assigns a confidence level within which he or she expects the completion time to fall based on the selected confidence level. For example, if user wants to be accurate 95% the time then he or she needs
20 choose a 95% confidence level. The number (standard deviation multiplier) for the 95% confidence level will be 1.96 considering standard normal distribution. In this case, all the completion times for the jobs will be calculated using z value of 1.96. If user want to evaluate different uncertainty values, he or she simply can change the confidence level.

 While the present invention has been described in connection with the preferred
25 ~~embodiments of the various Figs., it is to be understood that other similar embodiments may be~~
~~used or modifications and additions may be made to the described embodiment for performing~~
the same function of the present invention without deviating therefrom. For example, one skilled in the art will recognize that the present invention as described in the present application may apply to any computing device or environment, whether wired or wireless, and may be applied to
30 any number of such computing devices connected via a communications network, and
interacting across the network. Furthermore, it should be emphasized that a variety of computer

platforms, including handheld device operating systems and other application specific operating systems are contemplated, especially as the number of wireless networked devices continues to proliferate. Still further, the present invention may be implemented in or across a plurality of processing chips or devices, and storage may similarly be effected across a plurality of devices.

- 5 Therefore, the present invention should not be limited to any single embodiment, but rather should be construed in breadth and scope in accordance with the appended claims.